

DOI: 10.5604/20830157.1194261

DEVELOPMENT ISSUES IN ALGORITHMS FOR SYSTEM LEVEL SELF-DIAGNOSIS

Victor Mashkov¹, Andrzej Smolarz², Volodymyr Lytvynenko³

¹University J.E. Purkyne, Usti nad Labem, Czech Republic, ²Lublin University of Technology, Institute of Electronics & Information Technology, Lublin, Poland,

³Kherson National Technical University, Department of Informatics & Computing Technology, Kherson, Ukraine

Abstract. The paper deals with the problem of developing probabilistic algorithm for system level self-diagnosis. The main goal of the suggested algorithm is to minimize the mean time of its executing. The algorithm is based on the computing of the posterior probability of fault-free state of each system unit. Final decision about unit's state is made on the chosen decision rule. The execution of the probabilistic algorithm is elucidated with the help of simple example and then explained for the case of more complex systems.

Keywords: complex systems, self-diagnosis, probabilistic algorithm, decision rule

PROBLEMY PROJEKTOWANIA ALGORYTMÓW AUTODIAGNOSTYKI NA POZIOMIE SYSTEMU

Streszczenie. Artykuł opisuje problem projektowania probabilistycznego algorytmu autodiagnostyki na poziomie systemu. Głównym celem proponowanego algorytmu jest minimalizacja średniego czasu wykonania. Algorytm oparty jest na obliczeniach prawdopodobieństwa a posteriori bezawaryjnego stanu każdej jednostki systemu. Decyzja o stanie jednostki podejmowana jest na podstawie wybranej reguły decyzyjnej. Działanie algorytmu probabilistycznego zostało opisane na prostym przykładzie, a następnie wyjaśnione dla przypadku bardziej złożonych systemów.

Słowa kluczowe: systemy złożone, autodiagnostyka, algorytm probabilistyczny, reguła decyzyjna

Introduction

In real complex systems, units are not necessarily homogeneous and can operate under different conditions. Therefore, units can have different levels of their reliability. This fact can be accounted for by assigning probabilities to the reliability of a unit.

Probabilistic approach to system level self-diagnosis [2] doesn't deal with such problems as t-diagnosability [1] and testing assignment [6]. Self-diagnosis which focuses on probabilities of fault-free and faulty states of system units is called probabilistic diagnosis.

The benefits of the probabilistic diagnosis are as follows:

- it is simpler than the system level self-diagnosis methods based on the PMC model [6],
- probabilistic algorithms are faster algorithms without restrictive assumptions on the testing assignment or on the fault sets (i.e., they do not place an upper bound on the number of permitted faulty units).

Among the first who investigated the probabilistic algorithms were H. Fujiwara and K. Kinoshita [3]. The probabilistic algorithms are based on the computing of the posterior probabilities of system unit states, upon which the decision about the states of the system units is made. The algorithm presented here aims at minimizing the mean time of its executing. For this, the algorithm is structured so that it has several branches.

1. Probabilistic algorithm

It is assumed that there is statistical information about the average time of branch executing, t_{Bi} . Each branch can be also assigned a probability that after its executing the states of all system units will be identified, P_{Bi} . (i.e., algorithm ends). At the beginning, the branch which corresponds to the most probable situation in the system is executed. If after executing of the first branch the states of system units are not identified, then there will be executed the second branch. The choice of the second branch is also made according to the same criterion. There is also a probability that the third branch will be used.

According to the described approach, the mean time of algorithm executing is equal to

$$M[t_{alg}] = t_{B1} + (1 - P_{B1})t_{B2} + (1 - P_{B1})(1 - P_{B2})t_{B3}$$

where t_{alg} – time of algorithm execution.

Branches of the algorithm correspond to the certain faulty situations in the system. The most probable is the situation when all system units are fault-free. The given situation corresponds to the

first branch of the algorithm. The second in order of probability is the situation when only one system unit is faulty. All the situations when the number of faulty units in the system is greater than 1 are considered as single situation. This combined situation corresponds to the third branch of the algorithm (see fig. 1)

Generally, while developing probabilistic algorithms it is assumed that test doesn't provide perfect fault coverage (i.e., testing fault-free unit not always detects a fault in the tested unit). But for the algorithm under consideration it is assumed that fault coverage is equal to 100%.

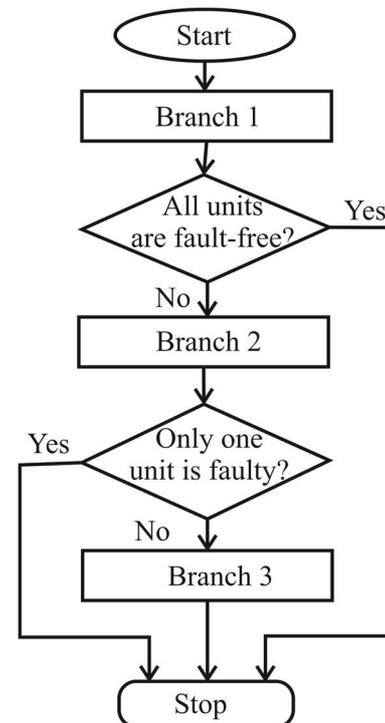


Fig. 1. Diagnosis procedure

Given such assumption, it is easy to conclude that the first branch of the algorithm consists in calculating the sum of test results which are equal to 1. If the resulting sum is equal to 0, then the algorithm ends with issuing information about fault-free states of all system units. Otherwise, the second branch is executed. It also consists in calculating the sum of test results which are equal to 1. But this time not all test results are taken into consideration.

Both the results of tests performed on i -th unit and the results performed by i -th unit are excluded. This i -th unit is determined on the basis of revising in sequence all the test results obtained after performing testing in the system. As soon as the test result which is equal to 1 is found, the revising procedure ends, and this test result (e.g., $r_{ji} = 1$) allows to determine the sought i -th unit.

Similarly to the first branch, if the resulting sum is equal to 0, the algorithm ends with issuing information about faulty state of unit u_i . Otherwise, the third branch is executed. It consists in computing the posterior probabilities of system units fault-free states. Having determined all the sought posterior probabilities, we can make decision about the state of each particular unit.

2. Simple example

Let's consider with a simple example how the posterior probabilities are determined. In this case, there are only two units u_1 and u_2 . Their prior probabilities that units are fault-free, p_1 and p_2 , are known. The probabilities of their faulty state are equal to $q_1 = 1-p_1$ and $q_2 = 1-p_2$ respectively.

Let's assume that there was performed only one test τ_{12} with the result $r_{12} = 0$ (see fig. 2).

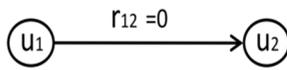


Fig. 2. Example with only one test

Now we are going to determine the posterior probabilities of fault-free state of unit u_2 taking into account the result of the performed test. Before performing test τ_{12} , there can be suggested the following four hypotheses: H_1 : units u_1 and u_2 are fault-free; H_2 : unit u_1 is faulty and unit u_2 is fault-free; H_3 : unit u_1 is fault-free and unit u_2 is faulty; H_4 : units u_1 and u_2 are faulty. The probabilities of these hypotheses are equal to: $P(H_1) = p_1p_2$; $P(H_2) = q_1p_2$; $P(H_3) = p_1q_2$; $P(H_4) = q_1q_2$.

Since hypotheses H_1, \dots, H_4 express a complete list of all possible states of the system (i.e., entire sample space), the sum of their probabilities is equal to 1, that is

$$\sum_{i=1}^4 P(H_i) = 1$$

Let A be the event when in the system there was performed the test τ_{12} with the result $r_{12} = 0$. Having made the assumption that $P_{AT} = 1$ (i.e., fault coverage of the test is equal to 100%) we can determine the conditional probabilities that event A occurs given hypotheses $H_i, i = 1, \dots, 4$, have been chosen (i.e., are true).

$P(A/H_1) = 1$; $P(A/H_2) = P_r$; $P(A/H_3) = 0$; $P(A/H_4) = P_r$; where P_r is the probability of obtaining zero result of the test performed by the faulty unit.

The probability of event A can be determined by using the formula for total probability

$$\begin{aligned} P(A) &= \sum_{i=1}^4 H(H_i)P(A/H_i) = \\ &= p_1p_2 + P_rq_1p_2 + P_rq_1q_2 = p_1p_2 + P_rq_1 \end{aligned}$$

Using Bayes' theorem, it is possible to determine the conditional probabilities of hypotheses $H_i, i = 1, \dots, 4$, given that event A occurs

$$P(H_i/A) = \frac{P(H_i)P(A/H_i)}{P(A)}, \quad i=1,4$$

$$P(H_1/A) = \frac{p_1p_2}{p_1p_2 + q_1P_r}; P(H_2/A) = \frac{p_1P_rq_1}{p_1p_2 + q_1P_r};$$

$$P(H_3/A) = 0; P(H_4/A) = \frac{q_1q_2P_r}{p_1p_2 + q_1P_r}$$

Unit u_2 can be recognised as fault-free when either hypothesis H_1 or hypothesis H_2 is true. Thus, the posterior probability that unit u_2 is fault-free can be determined as

$$P_2^* = P(H_1/A) + P(H_2/A) = \frac{p_1p_2 + q_1P_r p_2}{p_1p_2 + q_1P_r}$$

If we assume that the prior probabilities of fault-free state of units u_1 and u_2 take the same value and are equal to $P_1 = P_2 = 0.8$, then the posterior probability P_2^* will be equal to 0.973. From this example it follows that performing only one test with the result equal to 0 considerably increases the probabilities of the units fault-free states.

3. General case

Let's consider more complex example. Assume that the system consists of five units and its testing assignment is such as shown in fig. 3.

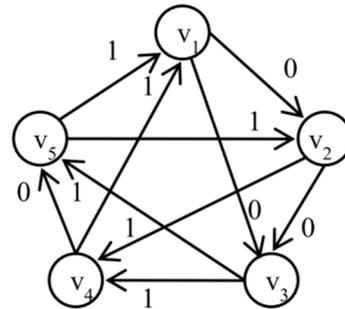


Fig. 3. Complex example

The results of tests are presented in the testing graph next to the corresponding edges. Let's also assume that the prior probabilities of fault-free states of the units are equal to $P_i = 0.8, i = 1, \dots, 5$.

According to the above presented method the following posterior probabilities of units fault-free states can be computed as $P_1^* = 0.883, P_2^* = 0.939, P_3^* = 0.941, P_4^* = 0.055, P_5^* = 0.008$.

Analysis of the values of the received probabilities allows to make decision that units u_1, u_2 and u_3 are fault-free, whereas the units u_4 and u_5 are faulty.

Usually, the decision about the states of system units is made on the basis of the chosen decision rule. There can be suggested simple likelihood ratio such as

$$\chi = \frac{P_i^*}{1 - P_i^*}$$

According to the above presented method, there can be developed the algorithm for determining the probability P_i^* for arbitrary testing graphs. The algorithm works as follows.

Given the testing graph $G(V, E)$ and actual syndrome R_F as algorithm input, the number of simple cycles with zero weights of edges, Z , is determined. For each simple cycle, C_Z , the number of vertices, L_Z , and the number of edges, K_Z , it contains is determined. Then, for each vertex of the Z -th cycle the posterior probability of fault-free state of the unit which corresponds to this vertex is determined by formula

$$P_i^* = \frac{\prod_{j=1}^{L_Z} p_j}{\prod_{j=1}^{L_Z} p_j + p_r^{K_Z} \prod_{j=1}^{L_Z} q_j},$$

where p_j, q_j – the prior probabilities of fault-free and faulty states of unit $u_j, \forall j: v_j \in C_Z$.

At the next step, all edges among the vertices which compose the cycle are excluded. After that, for each vertex the subset of vertices $\{V_i\}$ is determined, where

$$\{V_i\} = \Gamma_i^- / r_{ji} = 0, \quad v_j \in \Gamma_i^-$$

where Γ_i is the set of units tested by unit u_i and Γ_i^- is the set of units testing unit u_i .

Vertices which belong to the subset $\{V_i\}$ correspond to the system units which test unit u_i with the result equal to 0. Having determined the subset $\{V_i\}$, we can compute the probability P_i^* by the following formula

$$P_i^* = 1 - q_i^* \prod_{j/r_{ji}=0} (q_j^* p_r) \prod_{\substack{r_m=0/r_{ri}=0, \\ r_{mi}=0}} p_r$$

At the last step, all the rest tests which have the results $r_{ji} = 1$, $r_{ij} = 1$, $r_{ij} = 0$ are accounted, and the sought probability P_i^* is corrected. The probability P_i^* is determined for each system unit u_i , $i = 1, \dots, N$.

Having determined the posterior probabilities of fault-free states for all system units, we can decide upon which system units are indeed fault-free and which are faulty. This decision is made on the basis of the chosen decision rule.

For determining the posterior probabilities of system units fault-free states there were considered different hypotheses about the sets of faulty units in the system. However, it is worth noting that there can be obtained the syndrome R_F which doesn't correspond to any hypothesis. In this case, we say that there is a conflict situation. Presence of such situation indicates that there are intermittent faults in the system [4]. For diagnosing intermittently faulty system units special diagnosis algorithms should be applied [5].

4. Conclusions

Before developing diagnosis algorithm for system level self-diagnosis, we have to make decision on several issues. First of all it is necessary to choose the diagnosis strategy (i.e., perform unique diagnosis, or sequential diagnosis, or excess diagnosis). Next issue concerns the similarity of system units (i.e., either homogeneous or heterogeneous system). Finally, we need to make assumption about allowable faulty situations in the system (i.e., consider situations when either only permanent faults are allowable, or only intermittent faults are allowable, or hybrid faulty situation is allowable). Probabilistic algorithm suggested in the paper is developed for unique diagnosis of heterogeneous systems when only permanent faults are allowable. Correctness of the diagnosis performed with the probabilistic algorithms depends considerably on the chosen decision rule and on the made assumptions about test results (e.g., assumption about fault coverage of the test). In view of this the probabilistic algorithms are correct as long as the made assumptions are met.

References

- [1] Barsi T., Grandoni T., Maestrini P.: A theory of diagnosability of digital systems. IEEE Trans. on Comput., Vol. C-25, No. 6, 1976, 585–593.
- [2] Blount M. L.: Probabilistic treatment of diagnosis in digital systems. 7th IEEE Int. Symp. On Fault-Tolerant Computing, 1977, 72–77.
- [3] Fujiwara H., Kinoshita K.: Some existence theorems for probabilistically diagnosable systems. IEEE Trans. on Comp. Vol. C-27, No. 4, 1981, 297–303.
- [4] Mallela S., Masson G.: Diagnosable systems for intermittent faults. IEEE Trans. Comput., Vol. C-27, No. 6, 1978, 560–566.
- [5] Mashkov V. A., Mashkov O.A.: Diagnosis of sensor networks applied for environment monitoring. Int. Journal "Ecological Sciences". Vol. 7, No. 1, 2015, 120–130.
- [6] Preparata T., Metzger G., Chien R.: On the connection assignment problem of diagnosable system. IEEE Transactions on Electronic Computers. Vol. EC-16, No. 12, 1967, 848–854.

Prof. Victor Mashkov

e-mail: viktor.mashkov@ujep.cz

Doctor of Technical Sciences, Docent of Department of Informatics at the University of J.E. Purkyně in Usti nad Labem, Czech Republic. Research interests cover Dependability of information systems, fault-tolerance of computing systems, alliances and coalition formation.



Prof. Andrzej Smolarz

e-mail: a.smolarz@pollub.pl

Associate professor in Institute of Electronics and Information Technologies at Lublin University of Technology. Research field covers wide variety of optical methods in industrial diagnostics and control as well as applications of artificial intelligence methods in industrial diagnostics. Author of nearly 100 publications in this research area.



Prof. Volodymyr Lytvynenko

e-mail: immun56@gmail.com

Kherson National Technical University. DSc (Eng. hab.), Professor, Head of the Department Informatics & Computer Sciences. Research interests: artificial immune systems, time series forecasting, multifractal analysis, reinforcement learning.



otrzymano/received: 08.12.2015

przyjęto do druku/accepted: 30.12.2015